=======

SIMON'S BASIC

=======

SIMON'S BASIC

SIMON'S BASIC, C-64'ün emirleri arasına 90'dan fazla yeni emir katan bir yardımcı programdır. Şimdi bu emirleri, açıklamaları, örnekleri ile bazı dikkat edilmesi gereken hususları ile birlikte görelim.

Emir   : DIR
Format : DIR"string"

Disketin içinde bulunan programları görmek için yapılan LOAD"$",8 işlemi o anda bilgisayarda bulunan bir programı da siler. Ancak Simon's Basic'de bu soruna çözüm bulunmuştur. DIR"$" emrini vermek disketin fihristini görmek için yeterli olmaktadır. Liste bilgisayara doldurulmadan direk ekrana çıkarılır. Bu emri kullanmanın başka bir yolu da DIR"$:*=PRG", DIR"$:*=REL" veya DIR"$:*=SEQ", şeklinde kullanımıdır. Bu durumda sadece program, relavite veya sequential olan programlarınızın listesini görebilirsiniz. Veya DIR"$sim*" diyerek sadece "SIM" ile başlayan programların ekranda sıralanmasını sağlayabilirsiniz.

Emir   : DISK
Format : DISK"string"

Disk sürücüsüne emir göndermek için kullanılır. Normal Basic'te OPEN1,8,15,"emir":CLOSE1 şeklinde yapılan işlem sadece DISK"emir" yazmakla sağlanır. Diskete gönderilebilecek emirleri disk sürücüsünün el kitabında bulabilirsiniz. Çeşitli örnekler:

DISK"S:program*"-> İlk yedi harfi program olan tüm dosyaları siler.
DISK"N:DISK,01" -> Disketi "DISK" ismi ve 01 ID'si ile formatlar.
DISK"V"   -> Disketin üstündeki kullanılmayan hafızayı temizler.

Emir   : FETCH
Format : FETCH "string",uzunluk,değişken

Kontrollü bir şekilde INPUT yapmağı sağlar. Normal programlarda örneğin sayısal veri beklerken string verilmesi sonucu ortaya çıkabilen ERROR'ları engellemek ve INPUT sonucunda alınan cevabın belli bazı özelliklerde olmasını sağlamak için kullanılır. String kısmı, o verinin sahip olabileceği türü veya karakterleri saptar. Uzunluk ise verinin en çok kaç karakter olması gerektiğini belirtir. Değişkenin ise bir karakter değişkeni olması gerekir. Veri girişi sırasında sadece izin verilen karakterler ve INST/DEL tuşu kullanabilir. CURSOR tuşları çalışmaz. Son karakter verilince INST/DEL de çalışmaz.

String kısmında bulunan karakterlerin şu anlamları vardır.

(home   : Sadece satırda oluşturulabilecek karakterler.
(cursor down : Sadece sayı.
(cursor right : Sadece harfler ve SHIFT harfler.

Örnekler : FETCH"(home ",5,A$: Sadece ile z arasındaki harfleri kabul eder ve maksimum 5 karakter alır. Sonuç A$'a depolanır.
FETCH"12345",8,B$: Sadece 1,2,3,4 ve 5 sayılarından oluşan maksimum 8 karakterlik bir veri kabul eder.
FETCH"(home (down",5,C$: Tüm sayı ve harfleri alabilen en fazla 5 karakterlik veri alır.

Emir   : GRAPHICS
Fotmat : Değişken = GRAPHICS

GRAPHICS, "Video Interface Chip" denilen ve C-64'ün tüm grafiksel fonksiyonlarını düzenleyen kısmının ana adresini verir. Örneğin PRINT GRAPHICS denildiğinde 53248 sayısı elde edilir. Normalde POKE 53281,2 diyerek ekran rengini kırmızı yapmak mümkündür. Aynı şey POKE 53248+33,2 denilerek de yapılabilir. Veya POKE GRAPHICS+33,2 denilebilir. Bunun gibi çeşitli adresleri C-64'ün el kitabında bulabilirsiniz.

Emir   : INKEY
Format : Değişken=INKEY

INKEY o anda basılan fonksiyon tuşunun numarasını verir. Şu programı yazıp RUN edin ve çeşitli fonksiyon tuşlarına basın. Bastığınız tuşa göre 1 ile 15 arasında çeşitli sayıların ekrandan geçtiğini göreceksiniz. Hiçbirine basmazsanız 0 elde edersiniz. (1-8:normal/SHIFT s 16:C=/C=SHIFT)

```
10 PRINT INKEY
20 GOTO 10
```

Emir   : LIN
Format : Değişken=LIN

Cursor'un o anda bulunduğu satır numarasını verir. Bu sayı 0 ile 24 arasında olacaktır. Kullanılma alanı daha çok AT(x,y) emri ile olur. Örneğin PRINT AT(0,LIN+5)"Commodore 64" derseniz, Commodore 64 yazısı o anda geçerli olan satırın 5 satır altına yazılır.

Emir   : MEM
Format : MEM

DESIGN emri kullanılarak yapılacak karakter değiştirmelerinde en başta MEM emrinin kullanılması gerekmektedir. Bu emir bilgisayarın ekrana yazacağı karakterlerle ilgili bilgiyi içindeki ROM'dan değil de program lanabilir RAM'dan almasını sağlar. Böylece ekranda Türkçe harfler yazabilmek mümkün olmaktadır. Daha fazla bilgi DESIGN emrinde verilecektir.

Emir   : PAUSE
Format : PAUSE sayı veya PAUSE "string",sayı

Bilgisayar bu emri gördüğünde sayının belirttiği kadar saniye bekler ve bu arada herhangi bir string belirtilmişse bunu ekrana yazar. Böylece normal Basic'te FOR I=1 to 1000:NEXT gibi emirlerle yapılabilen boşluklar daha kolay programlanmış olur. <Return> ile kesebilirsiniz. Örneğin :

```
10 PRINT "20 SANIYE BEKLIYORUM"
20 PAUSE 20
30 PAUSE "SIMDIDE 10 SANIYE BEKLEYECEGIM",10
```

Dikkat edilmesi gereken şey PAUSE emri ile String arasında bir boşluk olması gerektiğidir. Örneğin PAUSE"BEKLE",10 emri hatalıdır.

Emir   : RESET
Format : RESET satır no

Normalde RESTORE emri, DATA'ları en baştan okumayı sağlar. Ancak RESET 30 emri bunu değiştirir ve 30. Satır sonra gelen DATA'ların yeniden okunmasını sağlar.

Emir   : SOUND
Format : Değişken=SOUND

Tıpkı GRAPHICS emri gibi bu seferde "Sound Interface Chip" diye tabir edilen ve C-64'ün tüm ses işlemlerini idare eden kısmının adresini verir. (=54272). Gerekli diğer bilgiler C-64 el kitabında vardır.

Emir   : OUT
Format : OUT

Bilgisayar tarafından en son verilmiş error yazısını yeniden verdirir.

Emir   : DISAPA ve SECURE
Format : SECURE sayı

Bu iki emir programdaki bazı satırları istenmeyen kişililerden saklamaya yarar. Öncelikle LIST edildiğinde görülmesi istenmeyen satırların başına DISAPA yazılır. Program yazımı bittikten sonra SECURE 10 emri verilerek bu satırların görünmez hale geçmesi sağlanır. Örnek :

```
10 DISAPA INPUT "PAROLAYI VERIN ";A$:IF A$<>"PAROLA" GOTO 10
20 PRINT "AFERIN,PAROLAYI BILDINIZ..."
30 DISAPA END
   SECURE 10
```

Emir   : AUTO
Format : AUTO sayı1,sayı2

Her satır yazımından sonra sayı2 kadar sonra gelen satır numarasının otomatik olarak verilmesini sağlar. Bu arada sayı1 ilk satır numarası olarak verilir. Örnek:

```
AUTO 45,6
45 PRINT "1.SATIR"
51 PRINT "2.SATIR"
```

Emir   : COLD
Format : COLD

Bilgisayarın içindeki her şeyi temizleyip SIMON'S BASIC'e yeniden başlamasını sağlar. Bu ara bir programınız silinirse OLD emri ile onu geri çağırabilirsiniz.

Emir   : DELAY
Format : DELAY sayı

LIST sırasında çeşitli yöntemlerle listeyi kontrol etme imkanı vardır. CTRL tuşu listenin geçmesini yavaşlatır. C= tuşu ise bastığınız sürece listeyi durdurur. SHIFT tuşu ise listeyi ekrana karakter karakter yazar. İşte DELAY emri bu sıradaki yazma hızını saptar. Sayı 1 ile 255 arasında olabilir ve 1 en hızlısıdır.

Emir   : DISPLAY
Format : DISPLAY

Fonksiyon tuşlarına key emrini kullanarak çeşitli anlamlar yüklemişseniz bu tuşların son durumunu görmek için bu emri veriniz.

Emir   : DUMP
Format : DUMP

Tüm değişkenlerin (matrisler hariç) o anki değerlerini gösterir.

Emir   : FIND
Format : FIND karakterler

Tüm program içinde bazı kelimelerin, örneğin bir değişkenin, hangi satırlarda geçtiğini görebilmek için bu emir kullanılır. Diyelim ki programdaki tüm GOTO'ları bulmak istiyorsunuz. FINDGOTO dediğinizde tüm GOTO'LU satır numaraları dizilir. Dikkat ederseniz FIND ve GOTO arasında boşluk yoktur. Çünkü boşluk koyarsanız, bilgisayar başında boşluk olan GOTO'ları arar.

Emir   : KEY
Format : KEY sayı,"string"

Fonksiyon tuşlarının programlanmasını sağlar. Örneğin KEY 1,"LIST" emiri F1'e LIST kelimesini yükler ve her F1'e basışınızda LIST yazılır. Eğer KEY 1,"LIST"+CHR$(13) yazarsanız bilgisayar LİST yazdıktan sonra otomatikman RETURN yapar ve programınız LİST edilir. (sayı1-16 arası olabilir).

Emir   : MERGE
Format : MERGE"program ismi",alet

Normal LOAD emri önce hafızadaki programı siler ve sonra yeni programı yükler. Ancak MERGE emri yeni programı eski programın arkasına ekler. Bu emirden önce bir OLD emri vermek bazı durumlarda ortaya çıkabilen sorunları giderir. Örnek :

MERGE"program",8 disketteki programı bilgisayardakine ekler.

Emir : OLD
Format : OLD
Yanlışlıkla NEW veya COLD emirleri verilince silinen programı yeniden geri çağırır.

Emir : OPTION
Format : OPTION 10 veya OPTION 0
LIST sırasında SIMON'S BASIC'e ait emirler eğer önceden OPTION 10 verilmişse revers olarak yazılır. OPTION 0 bunu durdurur.

Emir : PAGE
Format : PAGE sayı
Eğer bilgisayara PAGE 20 emri verilirse her LIST sırasında liste 20. Satıra geldiğinde durur ve kullanıcısının RETURN'a basmasını bekler. Sonra listeye ekranın ilk satırından devam eder. Böylece programınızı sayfa sayfa görebilirsiniz. Sayı 1-23 arasında olmalıdır. Daha büyükleri bilgisayarı normalde dondurur.

Emir : RENUMBER
Format : RENUMBER sayı1,sayı2
Programınızın tüm satır numaralarını sayı1'den başlayarak sayı2'lik adımlarla değiştirir. Bu arada GOTO, GOSUB, THEN, RESET gibi satır numaraları değiştirilmez.

Emir : TRACE
Format : TRACE 1 veya TRACE 0
RUN sırasında o anda işlenen satırın numarasını ekranın sağ üst köşesinde göstermeye yarar. Böylece programın işlemesi sırasında oluşabilecek mantık hatalarını görmek kolaylaşmaktadır.

Emir : PROC / ENDPROC / EXEC / CALL
Format : PROC isim / ENDPROC / EXEC isim / CALL
Normalde alt-programlar GOSUB emri ile çağrılır. Ancak SIMON'S BASIC'te her alt programa bir isim verilir ve CALL veya EXEC ile çağrılabilir. CALL alt-programa GOTO gibi gider, yani ENDPROC'u görünce geri dönmez. Halbuki EXEC o alt-programa gidince bir ENDPROC gördüğünde çağrıldığı yere geri döner. (GOSUB-RETURN gibi) bir örnek görelim:

```
 10 GOTO 100
 20 PROC COMMODORE
 30 PRINT "ALTPROGRAM BURASI !!"
 40 PRINT "COMMODORE 64"
 50 ENDPROC
 60 PRINT "ALTPROGRAM BITTI"
 70 END
100 FOR I=1 TO 10 : EXEC COMMODORE
110 NEXT I
```

Emir : ERRN veya ERRLN
Format : Değişken=ERRN veya değişken=ERRLN
En son errorun oluştuğu satır (ERRLN) veya errorun kodu (ERRN) bu değişkenlerin içine depolanır. Böylece ON ERROR kullanımı kolaylaşmış olur.

Emir : LOCAL ve GLOBAL
Format : LOCAL değişken1,değişken2,... ve GLOBAL

Normalde programın bir yerinde, örneğin bir alt-programda bir değişkenin içeriği değiştirildiğinde bundan tüm program etkilenir. Eğer alt-programda ilk satır olarak LOCAL emri verilirse ve çıkışta da GLOBAL emri yazılırsa programın diğer kısımları alt-programdaki işlemlerden etkilenmez. DİKKAT : LOCAL emri bir tek satırda yer almalıdır. LOCAL'den sonra iki nokta ile başka emirler yer almaz!!!

```
 10 A=5:C=3:A$="SIMON":B$=COMMODORE"
 20 PRINT A,C,A$,B$
 30 GOSUB 100
 40 PRINT A,C,A$,B$
 50 END
100 LOCAL A,A$
110 A=7:C=3:A$="64":B$="TRANSIT"
120 PRINT A,C,A$,B$
130 GLOBAL
140 RETURN
```

Emir   : IF...THEN...:ELSE:...
Format : IF start THEN emir1 :ELSE: emir2
IF ile THEN arasındaki start doğru ise THEN ile :ELSE: arasındaki emirler uygulanır. Eğer yanlışsa :ELSE:'den sonrası uygulanır.

Emir   : RCOMP...:ELSE:
Format : RCOMP emir1 :ELSE: emir2
Bir önceki satırdaki IF...THEN emrinin şartı doğru ise hem THEN'den sonraki işlemler hem de RCOMP ile :ELSE: arası yapılır. Yanlışsa :ELSE:'den sonrası gerçekleşir. Genellikle tek satıra sığmayan IF...THEN emirlerini birden fazla satıra yaymak için kullanılır. Örnek:

```
10 IF (((A(1)AND((AA()9)OR(A$="XX(")) AND B=1) OR C=3) THEN PRINT "ILK";
20 RCOMP PRINT"SATIRA DIGMADIGI ICIN BU SATIRA YAZILDI":ELSE:PRINT"YANLIS"
```

Emir   : LOOP...EXIT IF...END LOOP
LOOP ile END LOOP arası EXIT   IF'in arkasındaki şart oluşuncaya kadar tekrarlanır. Bundan sonra END LOOP'un arkasındaki satırdan devam edilir. DİKKAT : END ile LOOP arasında boşluk var !!!

```
 10 LOOP
 20 PRINT "X=";X
 30 X=X+1
 40 EXIT IF LOOP
 50 END LOOP
 60 PRINT"LOOP BITTI, X=";X
```

Emir   : ON ERROR
Format : ON ERROR: GOTO satırına
Bir hata oluştuğu sırada error verip durmak yerine satır no'ya gitmeyi sağlar. ERRN error kodunu, ERRLN error'lu satırı verir. OUT emri ise error yazısını vererek programı durdurur. ON ile ERROR arasında boşluk vardır!!!

```
 10 ON ERROR:GOTO 100
 20 FOR I=10 TO -1 STEP -1
 30 PRINT "KOK ";I;"=";SQR(I)
 40 NEXT 1:END
100 NO ERROR
110 PRINT"HATALI HESAPLAMA! SATIR : ";ERRLN
120 END
```

Emir   : NO ERROR
Format : NO ERROR
ON ERROR'un etkisinin kaldırır. NO ile ERROR arası boşluk vardır.

Emir   : REPEAT...UNTIL

REPEAT ile UNTIL arası, UNTIL'den sonra gelen şart oluşuncaya kadar tekrarlanır.

```
10 REPEAT
20 X=X+1
30 PRINT "X=";X
40 UNTIL X>20
```

GRAFİK EMİRLERİ İLE İLGİLİ AÇIKLAMALAR :

Her emrin arkasında görülecek MODE sözcüğü şu anlama gelmektedir: SIMON'S BASIC'le çizilen her çizgi değişik etkilere yol açabilir. Bunu MODE yerine konulacak sayı belirler.

```
Normal (320 X 200) grafik:0 :geçtiği yeri siler.
                          1 :geçtiği yere nokta koyar.
                          2 :geçtiği yerde nokta varsa siler yoksa koyar.
Multicolor (160 X 200)    0 :geçtiği yeri siler.
                          1 :geçtiği yeri 1 numaralı renkle boyar.
                          2 :geçtiği yeri 2 numaralı renkle boyar.
                          3 :geçtiği yeri 3 numaralı renkle boyar.
                          4 :geçtiği yerdeki 0 numaralı rengi 3 numaralı
                             renge, 1 no'lu rengi 2 no'ya veya bunları
                             tersine çevirir.
```

Emir   : ANGL
Format : ANGL merkez x, merkez y, açı x yarıçapı, y yarıçapı, mode

Merkez koordinatı, x ve y yönünde yarıçapı belli elipsin (veya dairenin) belirtilen açıdaki yarıçapını çizer. Saat yönünde düşünürsek 12=>0 veya 360 derece, 3=>90 derece, 6=>180 derece, 9=>270 derece olur.

```
10 HIRES0,1
20 FOR I=120 TO 10 STEP -4
30 ANGL 160,100,I,90,90,1
40 PAUSE 3: HIRES0,1
50 NEXT
```

Emir   : BLOCK
Format : BLOCK solüst x,solüst y,sağalt x,sağalt y,mode

Sol üst ve sağ alt köşelerinin koordinatları verilen dörgen alanın içini boyalı olarak oluşturur.

Emir   : CHAR
Format : CHAR x,y,kod,mode,ölçek

Grafik ekranında istenen x, y koordinatına, verilen koda karşılık gelen karakter verilen ölçeğe göre y yönünde büyütülerek konur. Verilecek olan kod ASCII kodları değil, C-64 el kitabında belirtilen ekran POKE kodlarıdır. Örneğin Q=Q, A=1, B=2... gibi. Olacak ise karakterin boyuna olan büyüklüğünü belirtir. Sayı büyüdükçe karakter uzar. DIKKAT : karakter kodu 0-511 arasında olmalıdır.

```
 5 HIRES0,1:A=1
10 FOR I=1 TO 300 STEP 9
20 CHAR I,8,I,1,A:A=A+1
30 NEXT:PAUSE4
```

Emir   : CIRCLE
Format : CIRCLE merkez x, merkez y, x yarıçapı, y yarıçapı, mode

Merkezi ve her iki eksen boyunca çapı belli elips çizer. X ve Y yarıçapları birbirine eşitse bu elips bir daire olur.

```
10 HIRES0,1
20 FOR I=1 TO 100 STEP 3
30 CIRCLE 160,100,100,I,1
40 NEXT:PAUSE4
```

Emir   : CSET
Format : CSET sayı

Arkasından gelen sayıya göre değişik işlemler yapar:

0 : Büyük harf/Grafik işaretleri seçilir.
1 : Küçük harf/Büyük harf seçilir.
2 : Bir önceki grafiği silmeden grafik moduna geçer.

Bu emrin ilk iki çeşidi (0,1) C= ve SHIFT tuşlarına basılarak da yapılabilir. Üçüncü şekli ise HIRES emrine bir alternatif olarak kullanılır. HIRES emri önce ekranı siler ve sonra grafik ekranına geçer. Bu durum daha önceden hazırlanmış bir ekranın yok olmasına yol açacaktır. CSET 2 ise bunu engeller.

Emir   : DRAW ve ROT
Format : DRAW string,x,y,mode ve ROT açı,büyüklük

Bu iki emir SIMON'S BASIC'in en ilginç emirlerindendir. Bu iki emirle değişik şekiller önceden belirlenip bir dtring değişkenine depolanabilir ve ekranın değişik yerlerinde 45 derecenin tam katları olarak döndürülebilir veya boyutları değiştirilebilir. Şimdi bu emrin ayrıntılarını görelim : En önemli kısım DRAW emrindeki string kısmıdır. Bu string kısmında kullanılan 0-9 arasındaki sayılar değişik işlemler yapar. Bu da istenen şekli yapmakta kullanılır. Ancak en fazla 255 karakterden oluşması gerekmektedir. Şimdi grafik ekranında küçük bir grafik cursor'u olduğunu düşünelim. Sayıların herbiri bu cursor'u şu şekilde etkiler:

0 - Sağa git
1 - Yukarı çık
2 - Aşağı in
3 - Sola git
4 - Bir anlamı yok
5 - Sağa git ve bir nokta bas
6 - Yukarı çık ve bir nokta bas
7 - Aşağı in ve bir nokta bas
8 - Sola git ve bir nokta bas
9 - Dur.

Her string'in son karakterinin 9 olması bazı sorunları ortadan kaldırır. ROT emri ise DRAW emri ile x, y koordinatlarına yerleştirilecek şeklin dönme açısını ve büyüklüğünü saptar. Burada açı 0-7 arasında bir sayı ile verilir ve şu anlamlara gelir :

0 - 0 derece	4 - 180 derece
1 - 45 derece	5 - 225 derece
2 - 90 derece	6 - 270 derece
3 - 135 derece	7 - 315 derece

Örnek program :

```
4 HIRES 0,1
5 A$="555555557888888875785555578888875785555555788888889"
6 FOR I=0TO7:ROT I,(1+2)/2:DRAW A$,100,I,20,1:NEXT
7 FOR I=0TO7:ROT I,(1+2)/2:DRAW A$,200,I,20,1:NEXT
8 PAUSE 4
```

Emir   : HIRES
Format : HIRES çizgi rengi,ekran rengi

Grafik ekranını siler, çizgi ve ekran rengini saptar ve grafik moduna geçer.

Emir   : MULTI
Format : MULTI renk1,renk2,renk3

Multicolor moduna geçmenizi sağlar. Önceden HIRES emri verilmelidir (veya CSET2). Multicolor modunda her 8x8'lik matriste 4 renk kullanma imkanı vardır. Bunun karşılığında x yönündeki nokta sayısı 320'den 160'a düşer. Bu nedenle MULTI emrinden sonra gelen tüm grafik emirlerindeki x koordinatlarını 2'e bölerek yazmak gerekir.

Emir   : LOW COL
Format : LOW COL renk1,renk2,renk3
HIRES veya MULTI ile belirlenen çizgi renklerini değiştirmeye yarar. LOW ile COL arasında boşluk var.

Emir   : HI COL
Format : HI COL
LOW COL ile değiştirilmiş renkleri tekrar HIRES veya MULTI ile belirlenmiş renklere dönüştürür. HI ile LOW arasında boşluk var.

Emir   : LINE
Format : LINE x1,y1,x2,y2,mode
X1, Y1 koordinatları ile X2, Y2 koordinatları arasına çizgi çizer.

Emir   : PAINT
Format : PAINT x,y,mode
X, Y koordinatları etrafında kalan kapalı alanı doldurur. Örnek :

```
10 HIRES 0,1
20 LINE 0,0,319,199,1: CIRCLE 160,100,100,90,1
30 PAINT 155,105,1
40 PAUSE
```

Emir   : PLOT
Format : PLOT x,y,mode
X ve Y ile belirlenen yere nokta koyar.

Emir   : REC
Format : REC solüst x,solüst y,x uzunluğu,y uzunluğu,mode
Sol üst köşe noktası ve kenar uzunlukları verilen dörtgeni çizer.

Emir   : TEST
Format : Değişken=TEST(x,y)
X, Y koordinatında nokta olup olmadığını kontrol etmek için kullanılır. Eğer varsa sonuç 1 olur; yoksa 0'dır.

Emir   : CMOB
Format : CMOB renk1,renk2
Multicolor'da çizilmiş Sprite'nın rengini değiştirir.

Emir   : TEXT
Format : TEXT x,y,"string",mode,y ölçeği,x ölçeği

Grafik ekranının istenen yerine string'i yazar. X ölçeği arka arkaya gelen iki harfin sol kenarları arasındaki nokta sayısını verir. Normalde 8 olmalıdır. Daha küçük olursa harfler üst üste biner. Daha büyük sayılarda ise harflerin arası açılır. Y ölçeği ise CHER emrinde olduğu gibi harflerin boyunu saptar. String'in ilk harfi CTRL-A ise string büyük harf/grafik işaretleri modunda; CTRL-B ise büyük/küçük harf modunda yazılır. RVS ON ve RVS OFF işaretleri geçerlidir. Dtring'den sonra gelelerin sayı olarak verilmesi gerekir; değişken olmaz.

Emir   : DESIGN
Format : DESIGN kod,adres

Multicolor veya normal mode'da karakter veya sprite şeklini program içinde kolaylıkla oluşturabilmek için kullanılır. Bu işlemlerden hangisinin yapılacağını kod belirler.

0 : Normal sprite   (24x21)
1 : Multicolor sprite   (12x21)
2 : Normal karakter   (8x8)
3 : Multicolor karakter (4x8)

Bilgisayar bu emri gördükten sonra gelen 8 veya 21 satır boyunca yerleştirilmiş olan işaretlerden oluşan kodları, belirtilen adresten itibaren yerleştirir. Bu adres sprite dizaynı için blok numarası çarpı 64'tür. Karakter dizaynı içinse $E000+ekran kodu *8'dir. Blok'un ve ekran kodunun anlamları C-64 el kitabında vardır. Kullanılabilecek blok numaraları : 13-15,32-63,128-255

Karakter dizaynında $E800 temel adrestir. $E400 alınırsa revers karakterlerin temel adresi, $E800 alınırsa küçük/büyük harflerin temel adresi, $EC00 alınırsa revers küçük/büyük harflerin temel adresi bulunur. Alt satırlarda dizayn belirtilmelidir. Bu başında @ bulunan ve 24,12,8 veya 4 işaretten oluşan 21 veya 8 satırdan oluşur. İşaretler ise nokta,B,C,D'dir. Anlamları :

. : Boşluk
B : (Normalde) çizgi rengi, (multicolorda) 1 numaralı renk
    (Spirite içinse CMOB'daki 1.Renk)
C : 2 numaralı renk (sprite içinse MOB SET'teki renk)
D : 3 numaralı renk (sprite içinse CMOB'daki 2.renk)

Aşağıdaki örnek program C= ile C'ye basıldığında yazılan grafik işaretini önce normal görünüş için, sonra da revers görünüş için değiştirmektedir. Önce MEM emrini bir daha okuyun isterseniz.

```
10 MEM:DESIGN2,$E000+124*8
20 @..BBBB..
21 @.BB..BB.
22 @.BB.....
23 @.BB.....
24 @.BB..BB.
25 @..BBBB..
26 @...BB...
27 @..BB....
30 DESIGN2,$E400+124*8
31 @BB....BB
32 @B..BB..B
33 @B..BBBBB
34 @B..BBBBB
35 @B..BB..B
36 @BB....BB
37 @BBB..BBB
38 @BB..BBBB
```

Emir : MOB SET
Format : MOB SET no,blokno,renk,öncelik,multi/normal
No ile belirtilen sprite'ın hangi blokta olduğunu, rengini, ekrandaki karakterlere göre önceliğini (1=>arkada, 0=>önde) ve hangi grafik modunda oluşması gerektiğini (0=>normal, 1=>multicolor) saptar. MOB ile SET arasında boşluk olmalıdır.

Emir : MMOB
Format : MMOB no,ilk x,ilk y,son x,son y,büyültme,hız
No'su belirtilen sprite'ı ilk x, Y'den son X, Y'ye belirtilen hızla götürür. 0 en hızlı, 255 en yavaştır. Ölçek ise sprite'ın hangi yönlere genişleyeceğini belirtir. (0:normal, 1:x yönünde, 2:y yönünde, 3: her iki yönde de)

Emir : MOB OFF
Format : MOB OFF no
No'su verilen sprite'ı görünmez yapar. MOB ile OFF arası boşluktur.

Emir : RLOCMOB
Format : RLOCMOB no,x,y,büyültme,hız
Sprite'ı bulunduğu yerden X, Y koordinatlarına götürür. Gerisi MMOB gibidir.

Emir : DETECT
Format : DETECT kod
Her CHECK emrinden önce ve sonra kullanılması gerekir. Kod 0 ise sprite-sprite çarpışması, 1 ise sprite-ekran çarpışması kontrol edilir.

Emir : CHECK
Format : CHECK (0) veya (no1,no2)
Bir önceki DETECT emrindeki koda göre ekranla sprite arasında (CHECK(0)) veya sprite no1 ile sprite no2 arasında (CHECK(no1,no2)) çarpışması olup olmadığını saptar. Çarpışma varsa sonuç 0 yoksa 1 olur.

Emir : AT
Format : AT(kolon,satır)
Verilen kolon, satırdan itibaren yazmayı sağlar. A$=at(5,5)+"COMMODORE 64" demekte mümkündür, ancak sadece program içinde yapılabilir. PRINT AT (5,5);"COMMODORE 64" hem program içinde hemde direkt emir vermede kullanılabilir.

Emir : CENTRE
Format : CENTRE liste
Aynı print emri gibi kullanılır. Özelliği arkasında gelenleri PRINT emri gibi satır başından değil de satırı ortalayarak yazmasıdır. Ancak yazma işlemini bitirdikten sonra otomatik (RETURN) yapmaz. Bu nedenle ekrana yazılacak bir şey hemen arkasına eklenecektir. Bu istenmiyorsa CENTRE emri kullanıldıktan sonra bir PRINT yazmanın yararı olacaktır.

Emir : DUP
Format : Değişken=DUP (string,sayı)
String'i sayı kadar çoğaltır. Örnek:

```
10 PRINT DUP ("A",10)
20 BB$=DUP("BALIK",5):PRINT BB$
```

Emir : INST
Format : Değişken=INST("string1","string2",yer)
String1'i belirtilen yerden itibaren string2'ye yerleştirir ve konduğu yerdeki string2'nin kısımlarını siler. Sadece program içinde kullanılabilir.

```
10 PRINT INST("XXX","COMMODORE 64",4)
```

Emir : INSERT
Format : Değişken=INSERT("string1","string2",yer)
String2'yi belirtilen yerden itibaren string1'e kadar ayırır ve oluşan boşluğa string1'i koyar. Sadece program içinde kullanılabilir.

Emir : PLACE
Format : Değişken=PLACE("string1","string2",yer)
String1'in, string2'nin içinde nerede olduğunu gösterir. Sadece program içinde kullanılabilir.

Emir : USE
Format : USE format,string
PRINT USING'in SIMON'S BASIC'teki kullanılış şeklidir. String kısmı string'e dönüştürülmüş sayıyı temsil eder. Bu STR$(sayı)şeklinde yapılabilir. Format kısmı ise şu özelliklere sahiptir:
1) # işareti sayıya denk gelmektedir. Nokta ise ondalık noktayı temsil eder. Örneğin 123,567 sayısı ###.## formatıyla 123.57 olarak yazılır.
2) ### işaretlerinin önüne veya arkasına değişik karakterler konulabilir
3) Eğer formatta nokta yerine başka bir şey kullanılırsa virgülden sonrası kesilir ve virgülden öncesi verilen formata uygun şekilde yazılır.
USE emri CENTRE gibi otomatik <RETURN> yapmaz.

Emir : DIV
Format : Değişken=DIV(sayı1,sayı2)
Sayı1'i sayı2'ye böler ve virgülden sonrasını atar.

Emir : EXOR
Format : Değişken=EXOR(sayı1,sayı2)
Sayı1 ile sayı2 arasında mantıksal EXOR işlemi yapar.

Emir : FRAC
Format : Değişken=FRAC(sayı1/sayı2)
Sayı1'i sayı2'ye böler ve virgülden öncesini virgülden sonrasını verir.

Emir : MOD
Format : Değişken=MOD(sayı1,sayı2)
Sayı1'i sayı2'ye böler ve kalanını verir.

Emir : $
Format : $ xxxx
Xxxx yerine gelen 4 basamaklı onaltılık sistemdeki sayıyı hesaplamaya katar.

Emir : %
Format : % xxxxxxxx
Xxxxxxxx yerine gelen 8 basamaklı ikilik sistemdeki sayıyı hesaplamaya katar. %00000011=3, %10000000=128 gibi. İkilik sistemde sayılar 0 ve 1 olabilir.

Emir : BCKGNDS
Format : BCKGNDS renk1,renk2,renk3,renk4
Extended Color tabir edilen mod'a geçilir. Bu modda her karakter 4 değişik zemin rengine sahip olabilir:
Renk1 : Normal zemin rengi. Renk2 : SHIFT-karakter'in zemin rengi
Renk3 : Revers-karakterin zemin zengi. Renk4 : SHIFT-revers-karakter'in zemin rengi.

Emir : BFLASH
Format : BFLASH hız,renk1,renk2 veya BFLASH 0
Ekran çerçevesinin verilen iki renk arasında, verilen hıza göre yanıp sönmesini sağlar (programdan bağımsız olarak). 1 en hızlı 255 en yavaştır. BFLASH 0 yanıp sönmeyi durdurur.

Emir   : COLOUR
Format : COLOUR çerçeve rengi, ekran rengi
POKE 53280,çerçeve rengi: POKE 53281,ekran rengi yerine geçer.

Emir   : COPY
Format : COPY
Grafik ekranını printer'a basar.

Emir   : DOWNB,DOWNW,LEFTB,LEFTW,RIGHTB,RIGHTW,UPB,UPW
Format : Emir satır,kolon,x-uzunluğu,y-uzunluğu
Satır ve kolon döndürülecek alanın sol üst köşesini belirtir. Bu emir normal yazı ekranında geçerlidir. Bu alan emrin gösterdiği yönde döndürülür. Eğer emrin arkasında B varsa boşalan yere boşluk yazılır. Dolayısı ile o alan bir süre sonra boşalmış olur. Eger W varsa örneğin UPW emrinde yukarıda kaybolan satır aşağıda ortaya çıkar.

Emir   : FCHR
Format : FCHR satır,kolon,x-uzunluğu,y-uzunluğu,kod
Sol üst köşesi ve uzunlukları verilen dörtgen alanı verilen ekran kodundaki karakterle doldurulur. Ancak daha önce bu alan boş ise yani bir renge sahip değilse dörtgenin bir kısmı görünmeyebilir.

Emir   : FCOL
Format : FCOL satır,kolon,x-uzunluğu,y-uzunluğu,renk
FCHR gibi ancak verilen alan belirtilen renkle doldurulur.

Emir   : FILL
Format : FILL satır,kolon,x-uzunluğu,y-uzunluğu,kod,renk
FCHR ve FCOL emirlerinin ikisinide aynı anda uygular.

Emir   : FLASH
Format : FLASH renk,hız
Ekrandaki karakterlerden verilen renge sahip olanların belirtilen hız la yanıp sönmelerini sağlar. 1 en hızlı, 255 en yavaştır.

Emir   : HRDCOPY
Format : HRDCOPY
Normal yazı ekranını Printer'e basar.

Emir   : INV
Format : INV satır,kolon,x-uzunluğu,y-uzunluğu
Yazı ekranında verilen alanı negatif hale getirir.

Emir   : MOVE
Format : MOVE satır1,kolon1,x-uzunluğu,y-uzunluğu,satır2,kolon2
Satır1, kolon1 ve uzunluklarla belirlenen alanın yeni yerde bir kopya sını çıkarır. Ancak dikkat edilmesi gereken şey yeni alanın ekrana tamamen sığması gerektiğidir.

Emir   : OFF
Format : OFF
FLASH emrinin etkisini kaldırır.

Emir   : SCRSV
Format : SCRSV kanal,alet,ikinci adres,"isim"
O anki yazı ekranını belirtilen alete (1=>kaset,8=>disket) verilen isim altında SAVE eder. DİKKAT kanal ne olursa olsun bu emir otomatik olarak CLOSE1 yapar.

```
10 SCRSV 2,8,2,"EKRAN1,S,W"
20 SCRSV 3,1,1,"EKRAN"
```

Emir : SCRLD
Format : SCRLD kanal,alet,ikinci adres,"isim"
ismi verilen ekranı, aletten ekrana geçirir. SCRSV gibi CLOSE1 yapar.

Emir : JOY
Format : Değişken=JOY
JOYSTICK 2'nin durumunu verir. 0 ile 8 arasında bir değer verir. Eğer aynı anda FIRE'a da basılıyorsa bu sayıya 129 eklenir.

Emir : PENX
Format : Değişken=PENX
LIGHTPEN'in o anki X-koordinatını verir.

Emir : PENY
Format : Değişken=PENY
LIGHTPEN'in o anki Y-koordinatını verir.

Emir : POT
Format : POT (no)
1.pot'taki PADDLE'in değerini verir. No ile hangi PADDLE'nin okunmak istendiği belirtilir (0 veya 1).

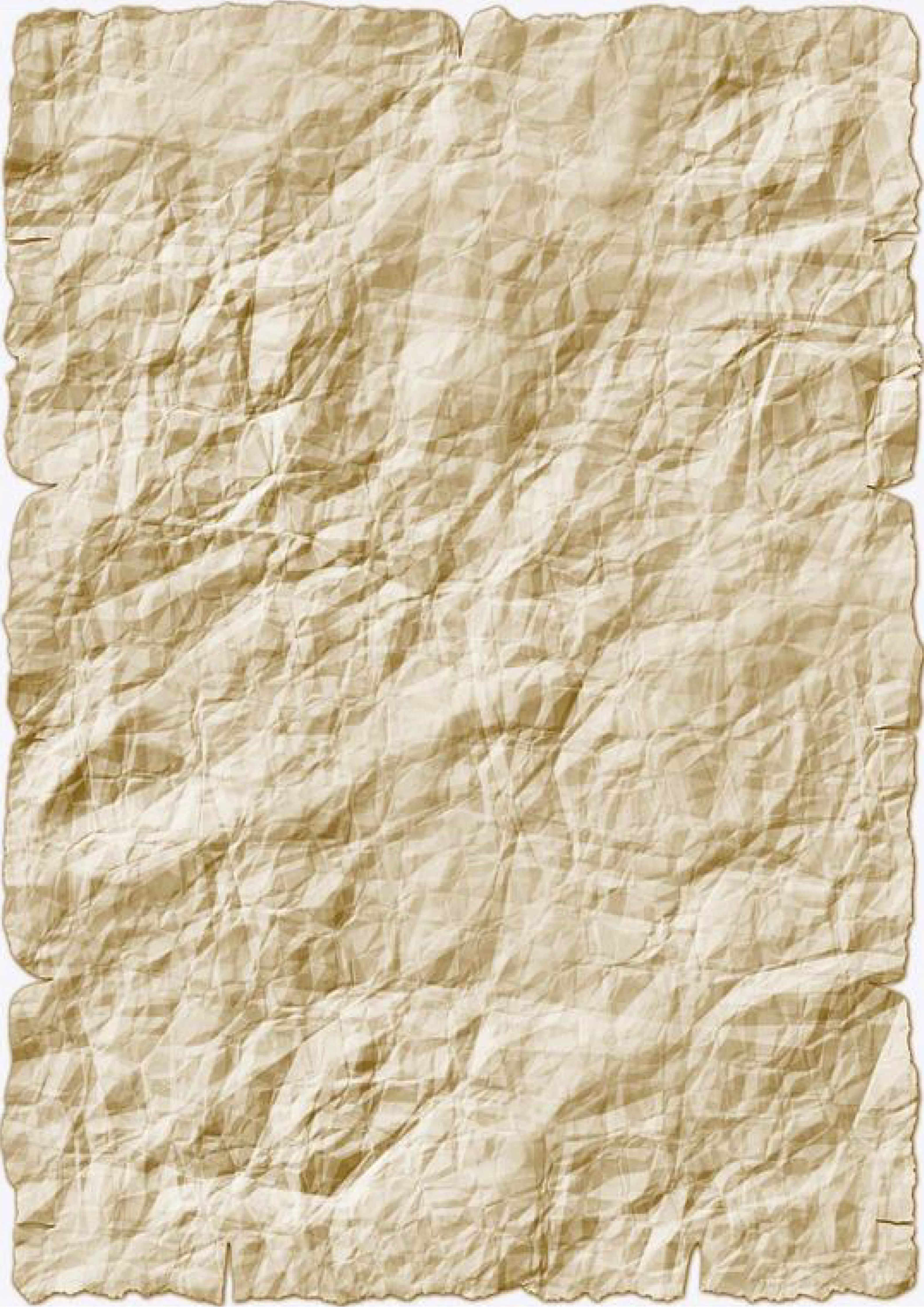
Emir : VOL
Format : VOL n
N ile belirtilen ses yüksekliğini ayarlar.

Emir : WAVE
Format : WAVE ses,ikilik sayı
Belirtilen sese (1-3), 8 basamaklı ikilik sistemde yazılmış sayının oluşturduğu sayı yazılır. Bu sayının oluşturduğu her bit'in bir görevi vardır ve şu anlamlara gelir:
0 : gate signal, 1 : syncranization, 2 : ring modulation, 3 : test bit
4 : üçgen dalga, 5 : testere dişi dalga, 6 : darbe/kare dalga, 7 : gürültü

Emir : ENVELOPE
Format : ENVELOPE ses,A,D,S,R
Belirtilen sesin zaman bağlı gidişini belirler. A,D,S,R ile ayrıntılı bilgiyi C-64'ün el kitabında bulabilirsiniz.

Emir : MUSIC
Format : MUSIC ölçü,string
Verilen string içindeki notaları verilen müzik ölçüsüne göre hafızasına yerleştirir. Stringi'in içindekilerin anlamları şöyledir :
(SHIFT+CLR/HOME) sayı: Sayının belirttiği sesi siler ve gelen notaları o ses çalar. Her string'in ilk karakteri olmalıdır.
F1 : 1/16'lık, F2 : 1'lik, F3 : 1/8'lik, F4 : 2'lik
F5 : 1/4'lük, F6 : 4'lük, F7 : 1/2'lik, F8 : 8'lik
Fonksiyon tuşları ile nota uzunluğu verilmeden önce Amerikan Standardına uygun şekilde nota ve oktav sayısı verilmelidir. Oktav sayısı 1-7 arasıdır. Nota : C = DO, D = RE, E = MI, F = FA, G = SOL, A = LA, B = SI
Her string (SHIFT+CLR/HOME)G ile bitmelidir.

Emir : PLAY
Format : PLAY mode
Daha önce MUSIC ile belirlenmiş parçanın çalmasını sağlar. Eğer mode1 ise parça çalarken program durur ve parçanın bitmesini bekler. 2 ise parça arka planda çalar ve program işlemeye devam eder. Eğer bu sırada çalan parçayı durdurmak isterseniz PLAY 0 emrini verin.



İLK BASKI : SKY ELEKTRONİK
REVİZYON TARİHİ : 01.04.2023
DÜZENLEYEN : RECEP MUM